

Title	PCG法の為のデータ構造(数値計算の基本アルゴリズムの研究)
Author(s)	戸川, 隼人
Citation	数理解析研究所講究録 (1985), 553: 100-107
Issue Date	1985-02
URL	<a href="http://hdl.handle.net/2433/98917">http://hdl.handle.net/2433/98917</a>
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

## P C G 法の為のデータ構造

日大・理工 戸川 隼人 ( Hayato Togawa )

PCG法は、差分法には使い易いが、有限要素法（特に不規則分割の場合）だとプログラムの作成に工夫を要する。そこで後者の為のデータの扱い方について検討した結果を報告する。

## 1. Band matrix 的な扱い方

分割がほぼ規則的で、一定のバンド幅と見なせる場合、剛性matrixの内部の非0区域もほぼ帯状になるので、その部分だけをメモリーに入れてband matrix的にPCG法を適用することができる。

例えば2次元問題の場合、非0帯は2本（対称行列の片側三角部分についての話、以下同様）になる。汎用プログラムの場合、非0区域の帯幅はわからないことが多いので要素データを自動的に調べて帯幅をもとめメモリをわりつける。

次のページに示すのは、そのようなプログラムの1例である。配列II (L, \*) は第L要素の節点番号一覧表である。それをもとに、

$$IND(K) = 0 \text{ or } 1$$

但しKは対角からのへだたりで、そこに0でない要素があれば

$$IND = 1, 0 \text{ でない要素がなければ } IND = 0 \text{ とする。}$$

という配列INDをつくる。INDの内容は、普通、

$$1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1$$

といった形になる。これを左から調べて、0になるまでを第1の帯、再び1になってから右端までを第2の帯とする。

その次のページからの4枚は、これを用いたPCG法の計算のプログラムで、初期設定、CG法の反復計算、不完全分解、代入計算、の順になっている。変数名はCG法の標準的な記法のとうり、例えばAPは係数行列Aと修正方向ベクトルpの積を表す。SIGMAは村田流の変形公式の $\sigma$ を表す。TUは解ベクトルである。

```

2650 N=NP
2660 DIM IND(N)
2670 FOR L=1 TO NE
2680     FOR IE=1 TO 8
2690         I=II(L,IE)
2700         FOR JE=I TO 8
2710             J=II(L,JE)
2720             IND(ABS(I-J))=1
2730         NEXT JE
2740     NEXT IE
2750 NEXT L
2760 J1=1
2770 FOR J=J1 TO N
2780     IF IND(J)=0 GOTO 2810
2790 NEXT J
2800 PRINT "band matrix テ`ナイ" : END
2810 J1=J
2820 FOR J=J1 TO N
2830     IF IND(J)=1 GOTO 2860
2840 NEXT J
2850 PRINT "アイタ`ニ スキマカ` ナイ" : END
2860 J1=J
2870 FOR J=J1 TO N
2880     IF IND(J)=0 GOTO 2910
2890 NEXT J
2900 PRINT "band matrix テ`ナイ" : END
2910 J1=J
2920 FOR J=J1 TO N
2930     IF IND(J)=1 GOTO 2960
2940 NEXT J
2950 PRINT "アイタ`ニ スキマカ` ナイ" : END
2960 J1=J
2970 FOR J=J1 TO N
2980     IF IND(J)=0 GOTO 3010
2990 NEXT J
3000 PRINT "band matrix テ`ナイ" : END
3010 J1=J
3020 FOR J=N-1 TO 1 STEP -1
3030     IF IND(J)=1 GOTO 3050
3040 NEXT J
3050 MB=J : BW=MB
3060 IF J1-1<MB THEN J1=MB+1
3070 M1=J1-1      ' タ`イ1ハ`ント`ハハ`
3080 M2=J1-J1     ' タ`イ2ハ`ント`ハハ`
3090 M3=J1-J1     ' タ`イ3ハ`ント`ハハ`
3100 M12=M1+M2
3110 M123=M1+M2+M3
3120 NW=N+MB
3130 K1=J1-J1     ' タ`イ1カンカクノ0ノコスウ
3140 K2=J1-J1     ' タ`イ2カンカクノ0ノコスウ
3150 K12=K1+K2
3160 PRINT "M1=";M1;": M2=";M2;": M3=";M3

```

```

10000 ' <<<<<<< PCG >>>>>>>
10010 *PCG
10020 PRINT "PCGホウ ノ ケイサン カイシ",TIME$
10030 GOSUB *DEC
10040 FOR I=1 TO N
10050     R(I)=TF(I)
10060 NEXT I
10070 PRINT : PRINT "シュッパ°ツチ ノ ケイサン ラ シテイマス"
10080 GOSUB *SOL
10090 FOR I=1 TO N
10100     TU(I)=SOL(I)
10110     R(I)=TF(I)-TK(K,0)*TU(K)
10120 NEXT I
10130 FOR K=1 TO N
10140     FOR J=1 TO M1 : L=K+J
10150         IF L>N GOTO 10290
10160         R(K)=R(K)-TK(K,J)*TU(L)
10170         R(L)=R(L)-TK(K,J)*TU(K)
10180     NEXT J
10190     FOR J=M1+1 TO M12 : L=K+K1+J
10200         IF L>N GOTO 10290
10210         R(K)=R(K)-TK(K,J)*TU(L)
10220         R(L)=R(L)-TK(K,J)*TU(K)
10230     NEXT J
10240     FOR J=M12+1 TO M123 : L=K+K12+J
10250         IF L>N GOTO 10290
10260         R(K)=R(K)-TK(K,J)*TU(L)
10270         R(L)=R(L)-TK(K,J)*TU(K)
10280     NEXT J
10290 NEXT K
10300 GOSUB *SOL
10310 FOR I=1 TO N
10320     P(I)=SOL(I)
10330 NEXT I
10340 S=0
10350 FOR I=1 TO N
10360     S=S+TF(I)*TF(I)
10370 NEXT I
10380 IF S=0 THEN RETURN ELSE EPS=N*S*1E-08
10390 RES=0
10400 FOR I=1 TO N
10410     RES=RES+R(I)*R(I)
10420 NEXT I
10430 S=0
10440 FOR I=1 TO N
10450     S=S+R(I)*P(I)
10460 NEXT I

```

```

10470 PRINT "ハンフク カイスウ", "サンサ 2ショウワ", "シヨク"
10480 SIGMA=S
10490 FOR KAISUU=1 TO 100
10500     PRINT KAISUU, RES, TIME$
10510     IF RES<EPS THEN RETURN
10520     FOR I=1 TO N
10530         AP(I)=TK(I,0)*P(I)
10540     NEXT I
10550     FOR I=1 TO N
10560         FOR J=1 TO M1 : L=I+J
10570             IF L>N GOTO 10710
10580             AP(I)=AP(I)+TK(I,J)*P(L)
10590             AP(L)=AP(L)+TK(I,J)*P(I)
10600         NEXT J
10610         FOR J=M1+1 TO M12 : L=I+K1+J
10620             IF L>N GOTO 10710
10630             AP(I)=AP(I)+TK(I,J)*P(L)
10640             AP(L)=AP(L)+TK(I,J)*P(I)
10650         NEXT J
10660         FOR J=M12+1 TO M123 : L=I+K12+J
10670             IF L>N GOTO 10710
10680             AP(I)=AP(I)+TK(I,J)*P(L)
10690             AP(L)=AP(L)+TK(I,J)*P(I)
10700         NEXT J
10710     NEXT I
10720     PAP=0
10730     FOR I=1 TO N
10740         PAP=PAP+P(I)*AP(I)
10750     NEXT I
10760     ALPHA=SIGMA/PAP
10770     FOR I=1 TO N
10780         TU(I)=TU(I)+ALPHA*P(I)
10790         R(I)=R(I)-ALPHA*AP(I)
10800     NEXT I
10810     GOSUB *SOL
10820     S=0
10830     FOR I=1 TO N
10840         S=S+R(I)*SOL(I)
10850     NEXT I
10860     BETA=S/SIGMA
10870     SIGMA=S
10880     FOR I=1 TO N
10890         P(I)=SOL(I)+BETA*P(I)
10900     NEXT I
10910     RES=0
10920     FOR I=1 TO N
10930         RES=RES+R(I)*R(I)
10940     NEXT I
10950 NEXT KAISUU
10960 RETURN

```

```

10970 ' <<<<<<< DEC(M1+M2) >>>>>>>
10980 ' version 1.0 (ハント" ハハ" M1+M2+1=M+1)
10990 *DEC
11000 PRINT "フカンセ"ン フンカイ ノ ケイサン ラ シテイマス"
11010 PRINT " キョウ ハンゴウ ";
11020 FOR I=1 TO N
11030     FOR J=0 TO M12
11040         LU(I,J)=TK(I,J)
11050     NEXT J
11060 NEXT I
11070 FOR K=1 TO N
11080     LOCATE 15,9 : PRINT K;
11090     P=LU(K,0)
11100     FOR J=1 TO M123
11110         UK(J)=LU(K,J)/P
11120     NEXT J
11130     FOR I=1 TO M1 : L=K+I
11140         IF L>N GOTO 11490
11150         Q=LU(K,I)
11160         IF Q=0 GOTO 11260
11170         FOR J=0 TO M1-I
11180             LU(L,J)=LU(L,J)-Q*UK(J+I)
11190         NEXT J
11200         FOR J=M1+1 TO M12-I
11210             LU(L,J)=LU(L,J)-Q*UK(J+I)
11220         NEXT J
11230         FOR J=M12+1 TO M123-I
11240             LU(L,J)=LU(L,J)-Q*UK(J+I)
11250         NEXT J
11260     NEXT I
11270     FOR I=M1+1 TO M12 : L=K+I+K1
11280         IF L>N GOTO 11490
11290         Q=LU(K,I)
11300         IF Q=0 GOTO 11390
11310         FOR J=0 TO M12-I
11320             IF J>M1 GOTO 11350
11330             LU(L,J)=LU(L,J)-Q*UK(J+I)
11340         NEXT J
11350         FOR J=M12+1 TO M123-I
11360             IF J>M12 GOTO 11390
11370             LU(L,J)=LU(L,J)-Q*UK(J+I)
11380         NEXT J
11390     NEXT I
11400     FOR I=M12+1 TO M123 : L=K+I+K12
11410         IF L>N GOTO 11490
11420         Q=LU(K,I)
11430         IF Q=0 GOTO 11480
11440         FOR J=0 TO M123-I
11450             IF J>M1 GOTO 11480
11460             LU(L,J)=LU(L,J)-Q*UK(J+I)
11470         NEXT J
11480     NEXT I
11490 NEXT K
11500 RETURN

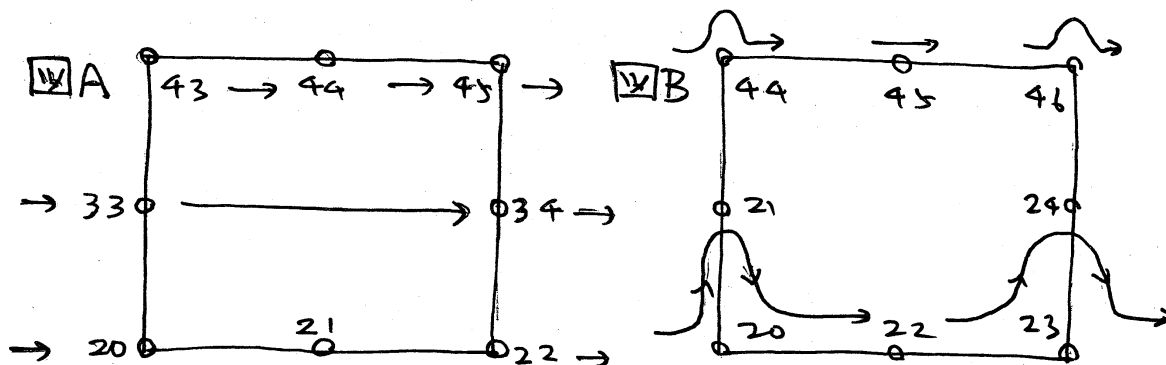
```

```

11510 ' <<< solve LU'*(LU~)*y=r >>>
11520 *SOL
11530 FOR I=1 TO N
11540     SOL(I)=R(I)
11550 NEXT I
11560 FOR K=1 TO N-1
11570     P=LU(K,0)
11580     SOL(K)=SOL(K)/P
11590     FOR J=1 TO M1 : L=K+J : IF L>N GOTO 11680
11600         SOL(L)=SOL(L)-LU(K,J)*SOL(K)
11610     NEXT J
11620     FOR J=M1+1 TO M12 : L=K+K1+J : IF L>N GOTO 11680
11630         SOL(L)=SOL(L)-LU(K,J)*SOL(K)
11640     NEXT J
11650     FOR J=M12+1 TO M123 : L=K+K12+J : IF L>N GOTO 11680
11660         SOL(L)=SOL(L)-LU(K,J)*SOL(K)
11670     NEXT J
11680 NEXT K
11690     SOL(N)=SOL(N)/LU(N,0)
11700 FOR I=N-1 TO 1 STEP -1
11710     S=0
11720     FOR J=1 TO M1 : L=I+J : IF L>N GOTO 11810
11730         S=S+LU(I,J)*SOL(I+J)
11740     NEXT J
11750     FOR J=M1+1 TO M12 : L=I+K1+J : IF L>N GOTO 11810
11760         S=S+LU(I,J)*SOL(L)
11770     NEXT J
11780     FOR J=M12+1 TO M123 : L=I+K12+J : IF L>N GOTO 11810
11790         S=S+LU(I,J)*SOL(L)
11800     NEXT J
11810     SOL(I)=SOL(I)-S/LU(I,0)
11820 NEXT I
11830 RETURN

```

このプログラムは、三角形1次式要素に関しては極めて有効であった。アイソパラメトリック要素（8節点長方形2次要素）にも使ってみたが、これは0区域が比較的狭いので、あまり効果がなかった。特に節接点番号を図Aのように付けるとよくない。バンド幅を狭くするにはAがよいが、0くいきを広くするには図Bのほうがよい。



## 2. ランダム・スパース行列の場合

非0要素を

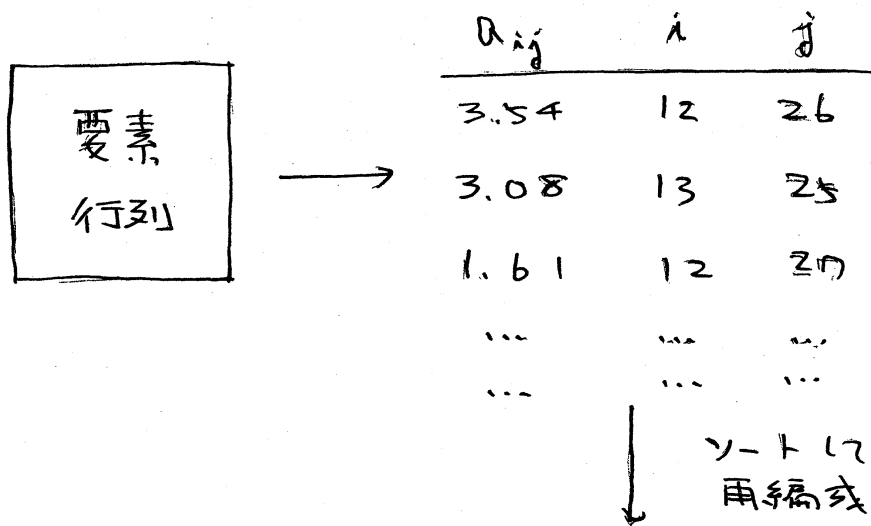
値	行番号	列番号
---	-----	-----

の組で表してファイルに入れ、まず行番号順にソートし、次に列番号順にソートし、同じ番号のものは加えあわせ、圧縮して計算用の配列にまとめ、これを用いてPCG計算を行う。ソートに少々時間がかかっても、このようにすれば

要素行列の重ねあわせが簡単になる

あとは1次元配列の形で処理できる

などの利点があり、全体としては時間短縮になる。



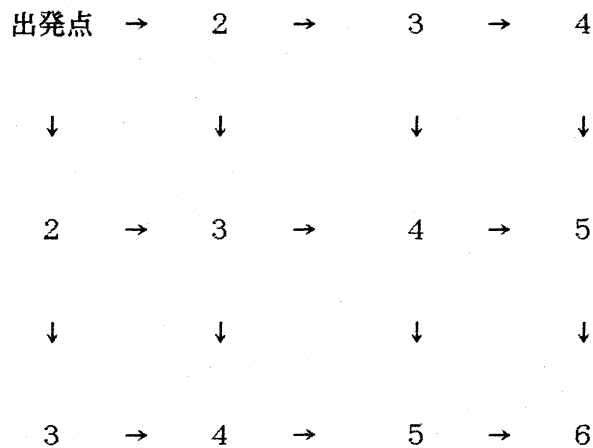
行番号	非0要素数	先頭位置	$a_{ij}$	$j$
1	3	1	2.86	1
2	1	4	0.19	2
3	5	5	1.54	7
⋮	⋮	⋮	3.111	2
			⋮	⋮

(7)



### 3. 並列処理できる場合

PCG法で、どこかの計算が並列処理できるかを考えてみると、ウェーブフロント法のフロント（先端の波紋）の上にある点の計算が並列処理可能であることがわかる。



（節点自由度が1より大きい場合は、ブロック的に扱う）

——これは昨年、後氏が提案したのと同じ。

### 4. 関連した話題

PCG法と同様にして、preconditioned Gauss Seidelとか、preconditioned SORなどを試みた。反復1回当たりの計算時間は少なくて済むが、必要反復回数が増すので、PCG法より不利になっている。

PCG法の番号付けをADI法のように2とおり併用することにより、情報伝達を速め、収束を速くすることを考えているが、まだ成功していない。

### 5. パソコンによる計算例

Laplace方程式の計算例を示す。PC-9801F, 8MHz, BASIC

(1) 有限要素法、三角形1次式要素、正方形領域、32\*32等分

節点数 2401    要素数 4608    バンド幅 50

所要時間 2時間20分

(2) 差分法（7点差分）、3次元、20\*20\*20分割

MICG(0)法、 所要時間 約6時間